

## Activity 1 : Public variables in the Editor

### Topics:

Variable scope in C# for Unity

Connection between public variables in script and the editor

Ranges for variables and their effect on editor interface

Changes in Play mode

### Goal:

Reinforcing the connection between unity editor GUI and the C# scripts, while exposing learners to additional tools that will segway into interface topics.

### Objective:

1. Identify a hard coded variable (int or float) that is suitable for changes during runtime.
2. Change the variable in script, then run the program to see it's effect.
3. Repeat the step above few times.
  - a. NOTE: The point of this is to allow students to recognize how much time is wasted changing and switching from IDE to Editor everytime a change is needed. Additionally, the changes can not be observed in play mode.
4. Notice that unless it is public, the variable is NOT accessible via the editor.
5. Ensure the variable is public.
6. Change the value of the variable in the editor during play mode.
7. Recognize that the value of the variable will not be preserved if it is edited during play mode.
8. During play mode, change the value of the variable to something "game breaking."
9. The range property allows for constraining variables of int and float types.
  - a. NOTE:  
<https://unity3d.com/learn/tutorials/topics/tips/show-public-variables-sliders-rangemin-max>
10. Whether working solo, or with others, it's good to constraint the range of values to only those that are valid and useful, especially since it only takes one line of code.
  - a. NOTE: This idea also transfers to other editor modification tools like named headers and serializedFields.

### Evaluation:

1. Understanding the difference between public vs private/protected scope in C# variables in Unity.
2. Recognizing the value of setting up the script/project for quicker changes, and changes during play mode.
3. Recognizing the value of variables changed during play mode will not be saved.
4. Ability to make changes to the script in order to change the value in editor.
  - a. Exposing the variable.
  - b. Locating the variable in the editor
  - c. Making changes in editor
5. Recognizing the existence of tools which can change how the editor looks / works and understanding the potential of using such tools.